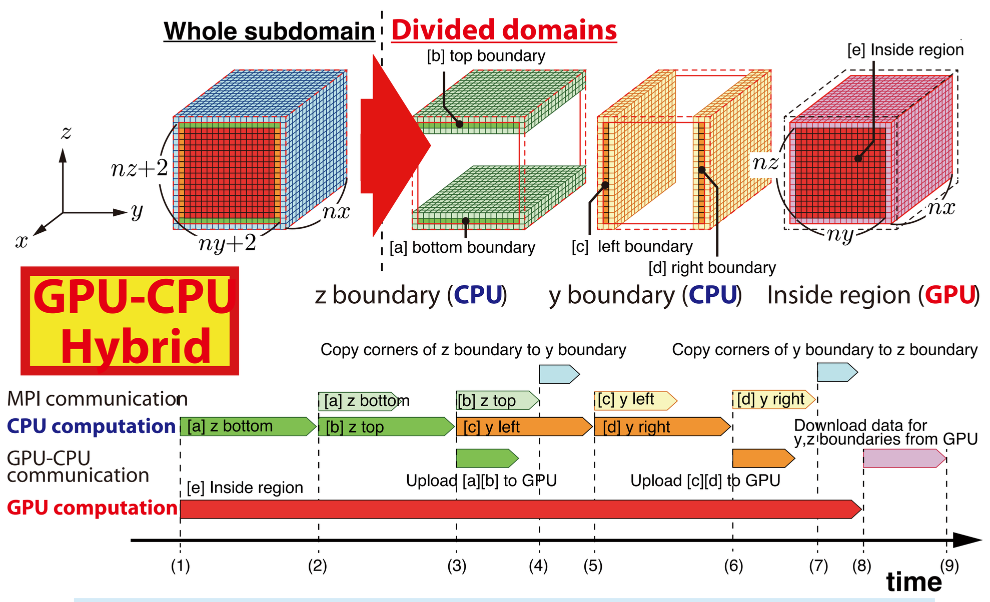
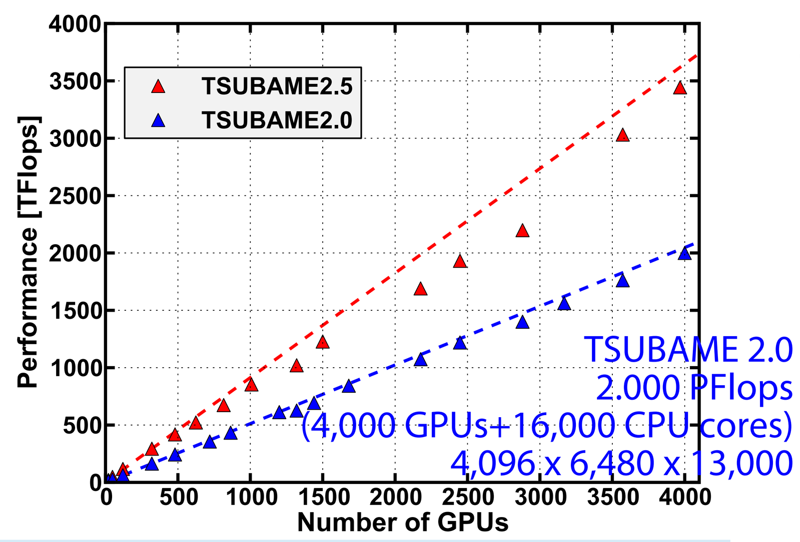
Peta-scale GPU Applications on TSUBAME

Phase-field simulation

The mechanical properties of metal materials largely depend on their intrinsic internal microstructures. The phase-field simulation is the most powerful method known to simulate the micro-scale dendritic growth during solidification in a binary alloy.

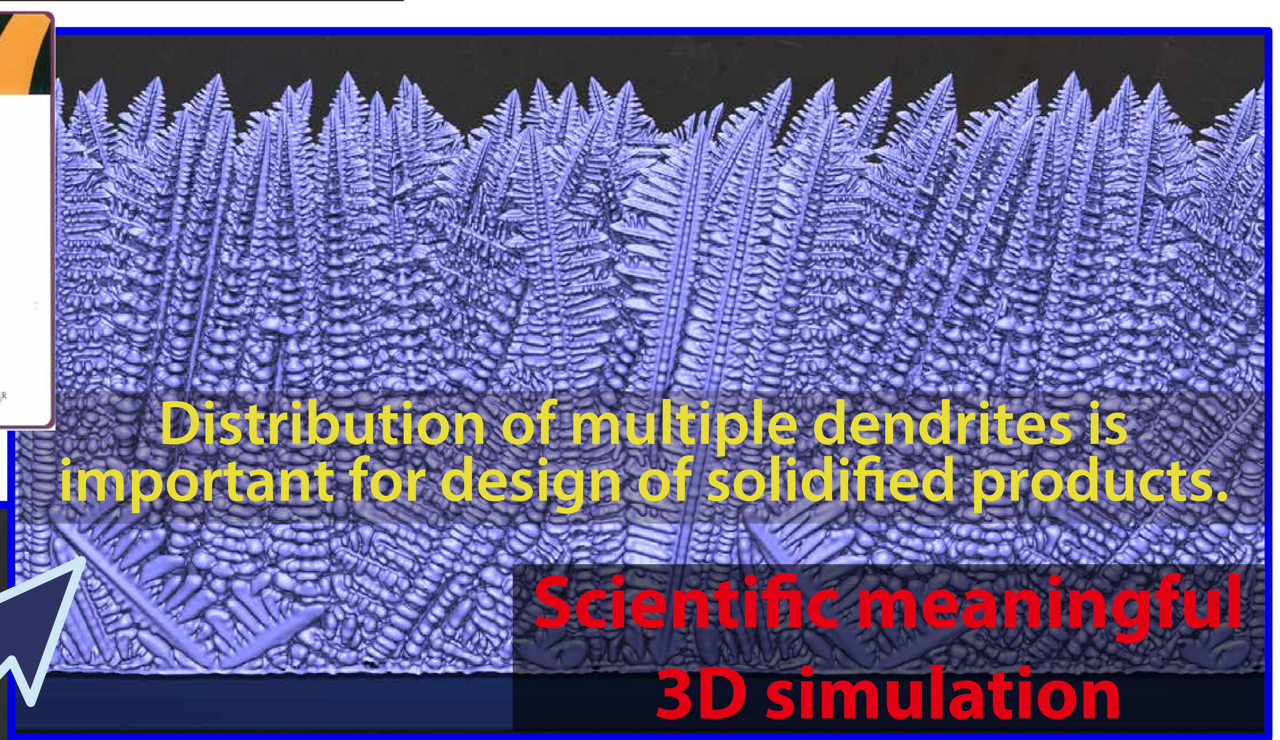


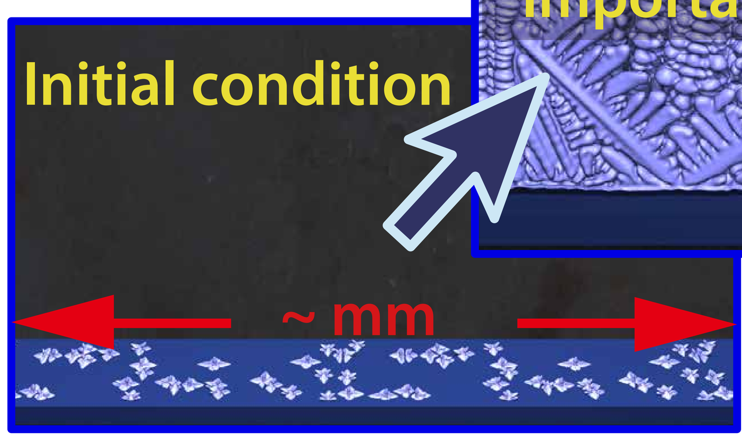
Scheme of the GPU-CPU Hybrid method



Weak scaling in single precision
Mesh size of a subdomain (1GPU + 4 CPU cores): 4096 x 162 x 130

2011 ACM Gordon Bell Prize
Special Achievements in Scalability and Time-to-Solution





Dendritic growth in the binary alloy solidification with 4096 × 1024 × 4096 (768 GPUs of TSUBAME2.0)

High-productivity framework for weather prediction code ASUCA

Skillful programming techniques are required for obtaining good parallel efficiency on GPU supercomputers. The Japan Meteorological Agency is developing a next-generation high-resolution meso-scale weather prediction code ASUCA. Our framework-based ASUCA has achieved good scalability with hiding complicated implementation and optimizations required for distributed GPUs, increasing the maintainability.

Stencil computation with the framework

- User-written function (C++ functor) that updates a grid point
- ArrayIndex3D represents the coordinate of the point where this function is applied.

Diffusion computation

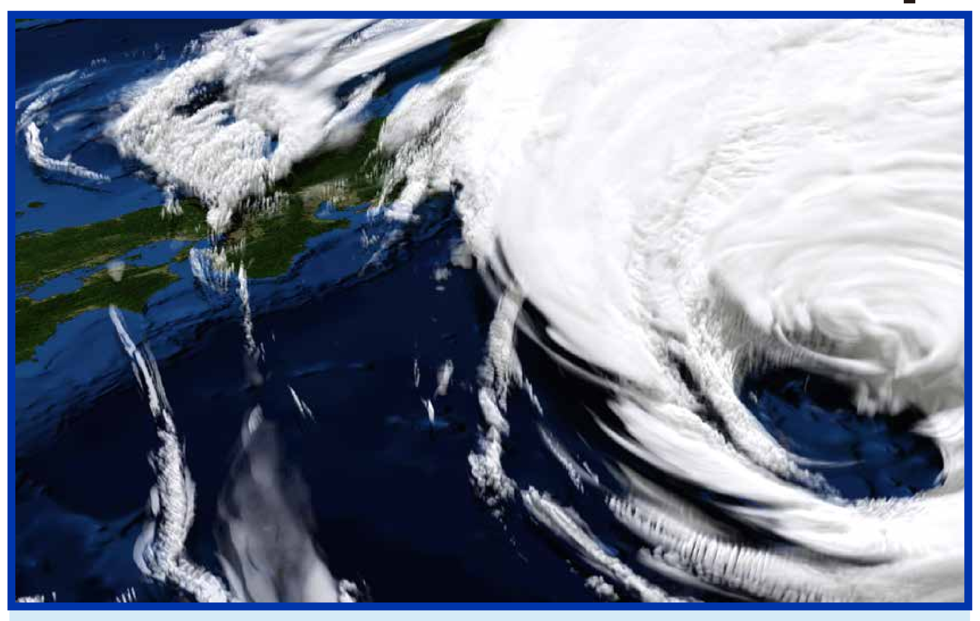
```
struct Diffusion3d {
  __host__ __device__
  void operator()(const ArrayIndex3D &idx,
    float ce, float cw, float cn, float cs, float ct, float cb, float cc,
    const float *f, float *fn) {
    fn[idx.ix()] = cc*f[idx.ix()] + ce*f[idx.ix<1,0,0>()] + cw*f[idx.ix<-1,0,0>()]
                 + cn*f[idx.ix<0,1,0>()] + cs*f[idx.ix<0,-1,0>()]
                 + ct*f[idx.ix<0,0,1>()] + cb*f[idx.ix<0,0,-1>()];
  }
};
```

- The functor is executed over all grid points by Loop3D provided by the framework.

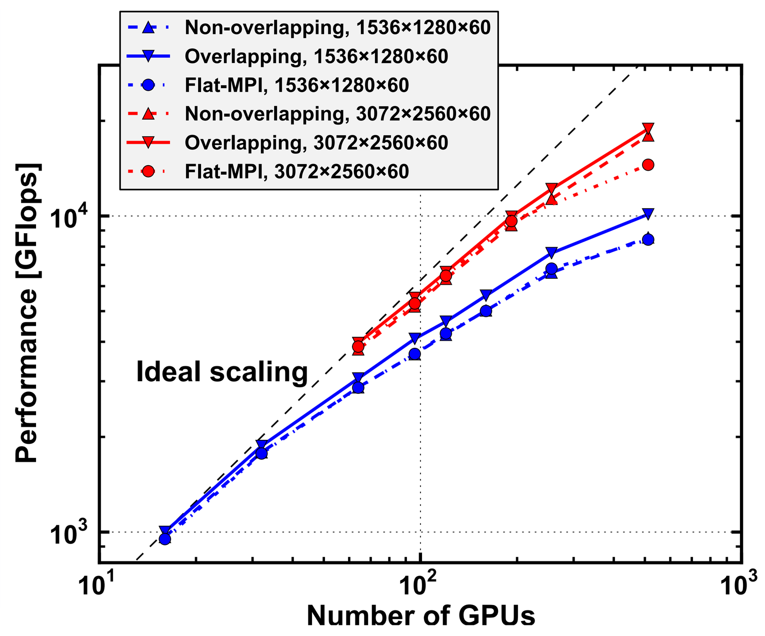
```
Loop3D loop3d(nx+2*mgnx, mgnx, mgnx, ny+2*mgny, mgny, mgny, nz+2*mgnz, mgnz, mgnz);
loop3d.run(Diffusion3d(), ce, cw, cn, cs, ct, cb, cc, f, fn);
```

User-written function   Parameters are provided to the user-written function.

Framework-based weather prediction code ASUCA



ASUCA real operation to describe a typhoon over Japan with 5,376 × 4,800 × 57 mesh using 672 GPUs of the TSUBAME 2.5.



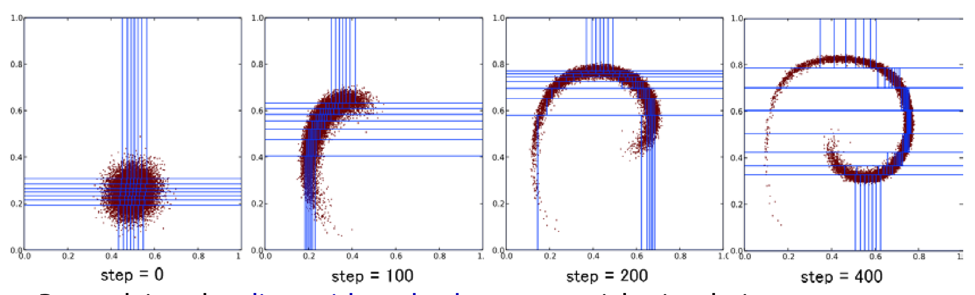
Strong scaling results of ASUCA.

Presentation at SC14: **Session:** Earth and Space Sciences
Time: 1:30-2:00PM on Tuesday, November 18th, **Room:** 391-92
Authors: T. Shimokawabe, T. Aoki and N. Onodera

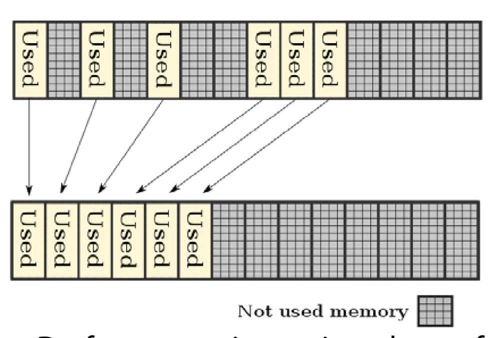
Particle simulation

The behavior of granular materials and the fluid dynamics can be simulated by using the particle method based on the short-range interactions such as DEM (Discrete Element Method) or SPH (Smoothed Particle Hydrodynamics). In order to bring the simulation closer to the real phenomena for the purpose of quantitative studies, it is necessary to execute large-scale particle-based simulations on modern high-performance supercomputers.

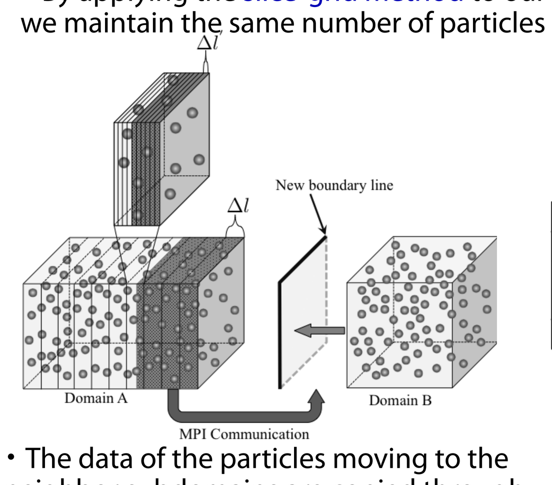
Dynamic Load Balance among GPUs



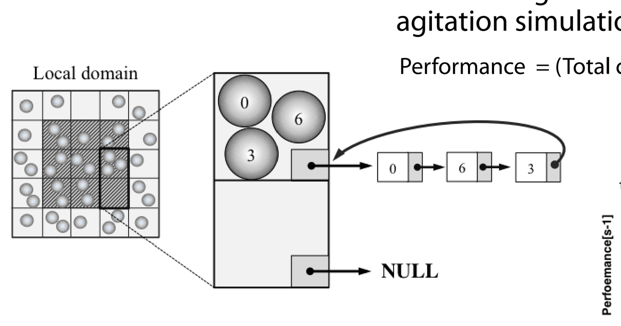
- By applying the slice-grid method to our particle simulation, we maintain the same number of particles in each domain.



- De-fragmentation revives the performance to linear increase.



- The data of the particles moving to the neighbor subdomains are copied through PCI-Express bus and CPUs memory.

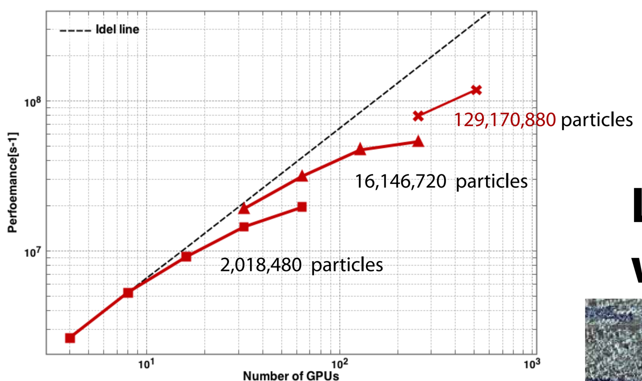


- We introduce the linked-list method for the neighbor particle list to save the memory drastically.

Performance scalability on TSUBAME

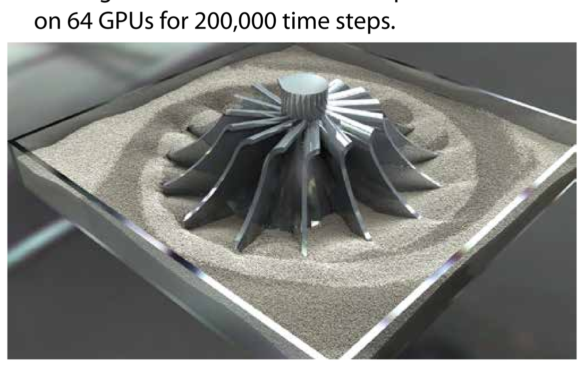
- The strong and weak scaling for the computation of agitation simulation are measured on TSUBAME 2.5.

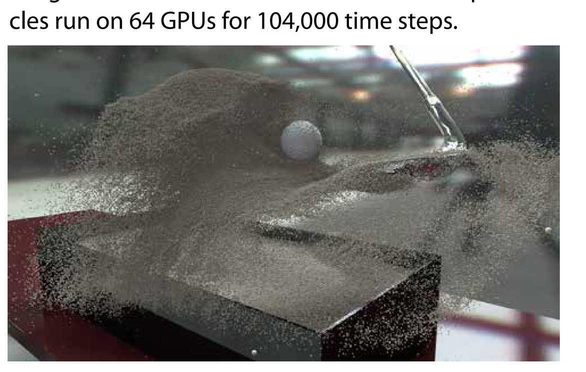
Performance = (Total computational time/Total step)$^{-1}$x Number of particles



Large-scale DEM simulations applied to the practical problems

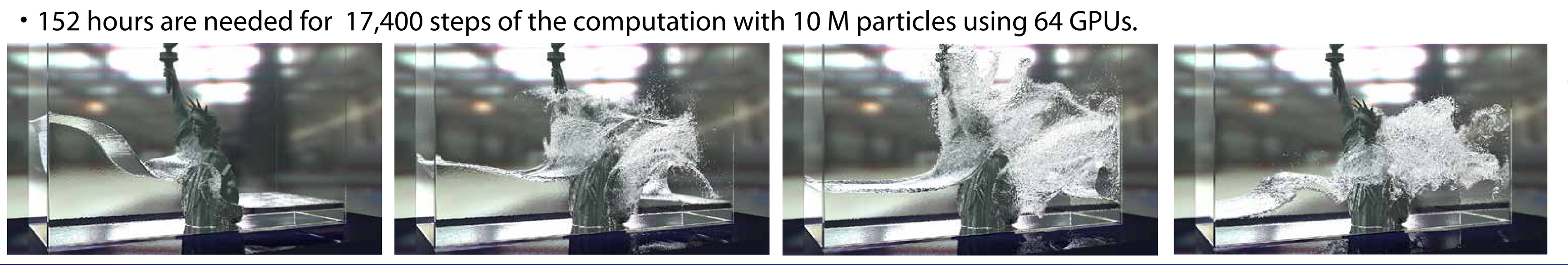
- A convey simulation with 4M particles run on 64 GPUs for 200,000 time steps.
- An agitation simulation with 4M particles run on 64 GPUs for 200,000 time steps.
- A golf bunker shot simulation with 16.7 M particles run on 64 GPUs for 104,000 time steps.





Application to the particle-based fluid simulations

- 152 hours are needed for 17,400 steps of the computation with 10 M particles using 64 GPUs.



LES wind simulation

The lattice Boltzmann method (LBM) is a class of CFD methods that solve the discrete-velocity Boltzmann equation. LBM continuously accesses memory with a simple algorithm and is suitable for large-scale computations including complicated objects.

LES lattice Boltzmann method

$$f_i(x + c_i\Delta t, t + \Delta t) = f_i(x,t) - \frac{1}{\tau}(f_i(x,t) - f_i^{eq}(x,t))$$

Relaxation time and eddy viscosity

$$\tau = \frac{3\nu_*}{c^2\Delta t} + \frac{1}{2} \qquad \nu_* = \nu_0 + \nu_{SGS}$$

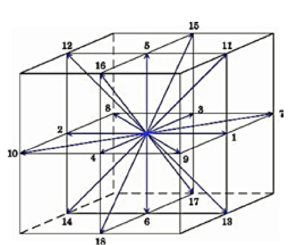
Coherent-structure Smagorinsky model

$$\nu_{SGS} = C|F_{CS}|^{\frac{3}{2}}\Delta^2|S|$$

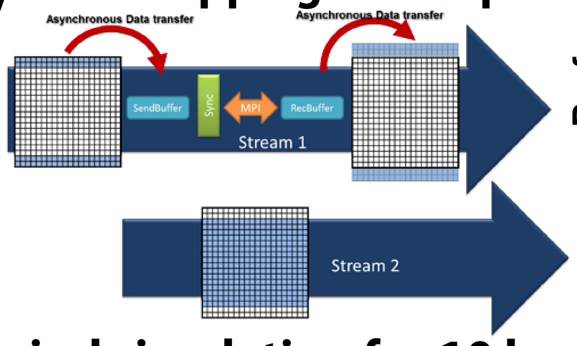
local memory access

$$F_{CS} = \frac{Q}{E} \qquad Q = -\frac{1}{2}\frac{\partial\bar{u}_j}{\partial x_i}\frac{\partial\bar{u}_i}{\partial x_j} \qquad E = \frac{1}{2}\left(\frac{\partial\bar{u}_j}{\partial x_i}\right)^2$$

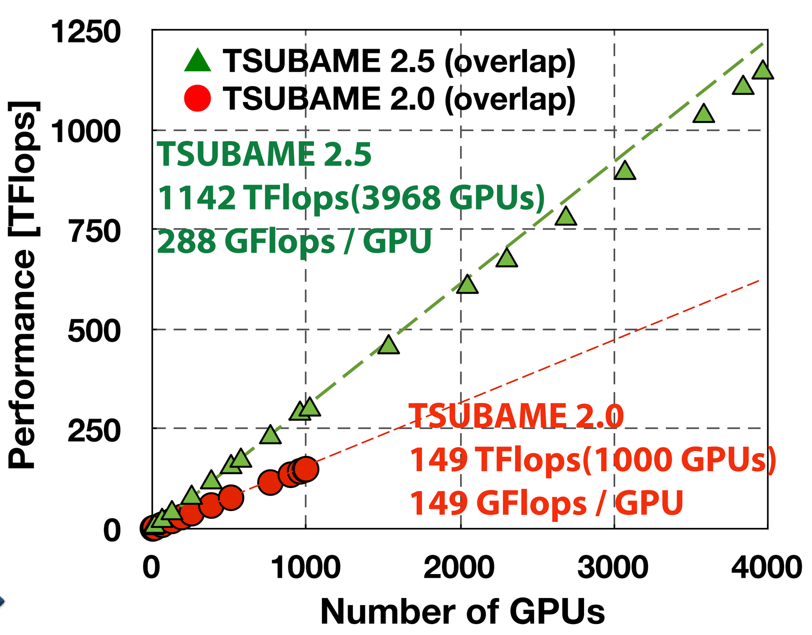
Discrete velocity
D3Q19 model



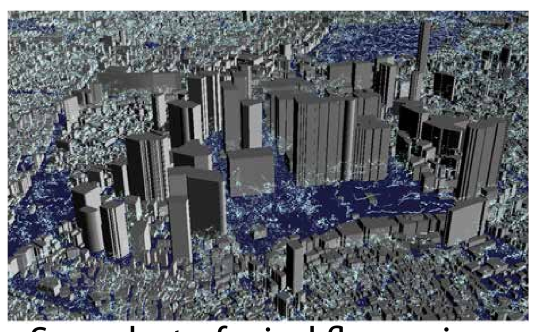
Overlapping technique



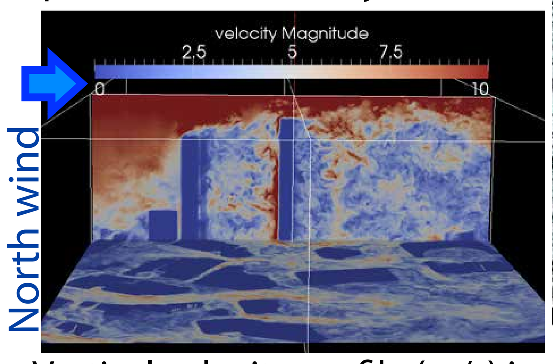
Weak scaling in single precision.
Mesh size of a subdomain: 192 x 256 x 256/GPU



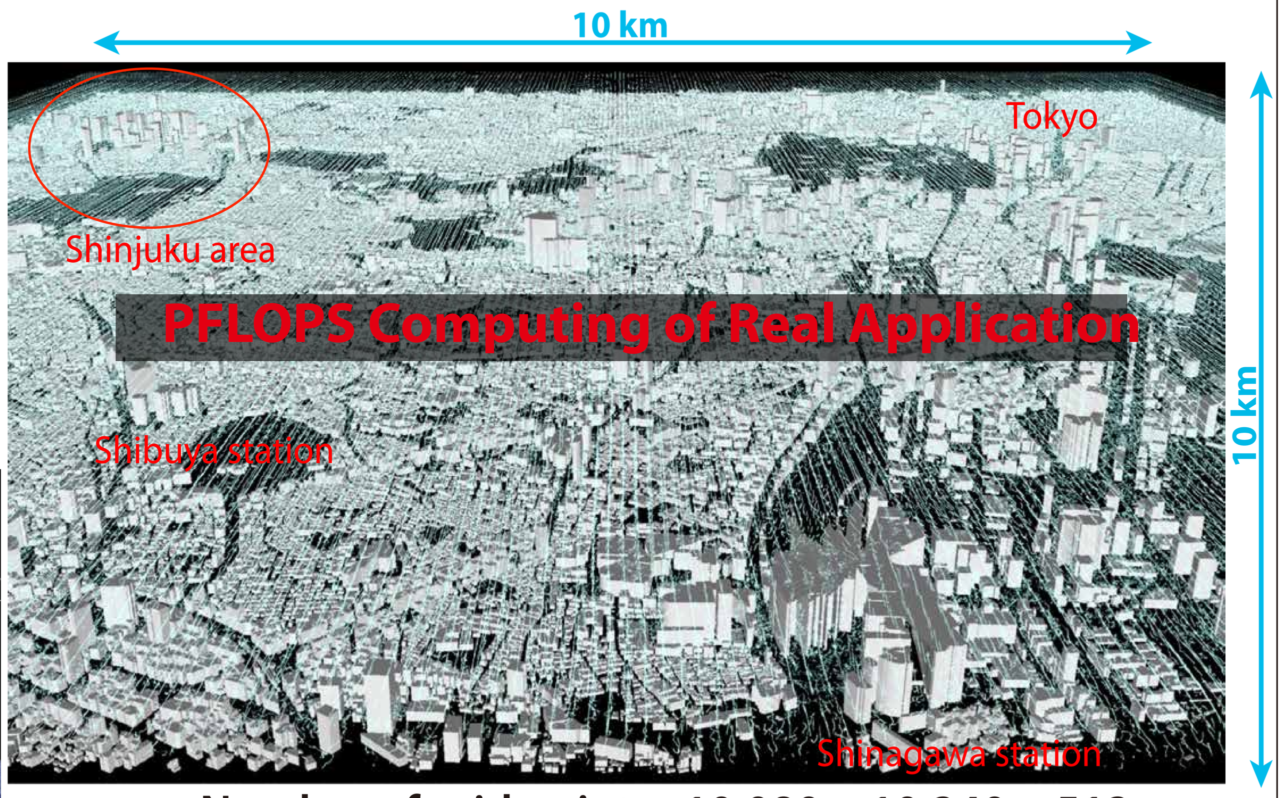
Large-scale LES wind simulation for 10 km x 10 km area in metropolitan Tokyo with 1-m resolution.



Snapshot of wind flow using particles in the Shinjuku area



Vertical velocity profile (m/s) in the Shinjuku area (north is left).



Number of grid points : 10,080 × 10,240 × 512 (4,032 GPUs of TSUBAME2.0)

http://www.gsic.titech.ac.jp/sc14